УДК 004.056.55
DOI: 10.12737/17147

Т.В. Карлова, Н.М. Кузнецова, А.Ю. Бекмешов

СОВЕРШЕНСТВОВАНИЕ СИММЕТРИЧНОГО ШИФРОВАНИЯ ПУТЁМ ВНЕДРЕНИЯ БЛОКА ИНФОРМАЦИИ ОБ ИСПОЛЬЗУЕМЫХ АЛГОРИТМАХ В КЛЮЧ

Рассмотрено применение нового подхода к шифрованию, основанного на внедрении в ключ блока дополнительной секретной информации о суперпозиции используемых открытых алгоритмов.

Ключевые слова: симметричное шифрование, алгоритмы шифрования, защита информации, информационная безопасность.

T.V. Karlova, N.M. Kuznetsova, A.Yu. Bekmeshov

SYMMETRIC ENCRYPTION UP-GRADE BY INTRODUCTION OF INFORMATION BLOCK ON USED ALGORITHMS IN KEY

The approach described in the paper is similar to the idea of memory organization in a computer. In the computer memory there must be saved not only data workable, but also directions for processing itself. Only in case when with encryption a key appears as a memory: it contains both, information essential in the course of encryption (a key proper), and an instruction on the way of encryption should be carried out (an additional block of information). As a consequence of this the crypto-durability of the whole system increases. At the same time the postulate of used algorithm openness is not broken: in the course of encryption there are used open algorithms only, only a key is secret which bears in itself the block of secret information on an open algorithm super-position.

Key word: Symmetric encryption, encryption algorithms, information security, information security.

Один из важнейших постулатов криптографии гласит: секретность криптосистемы должна быть основана на секретности ключа, а не на секретности алгоритма. В хорошей криптосистеме безопасность полностью зависит от ключа и абсолютно не зависит от знания алгоритма [1]. Известные криптографические алгоритмы, такие как TripleDES, RSA, AES, основаны именно на этом принципе. Однако криптоаналитикам в процессе вскрытия будет очень полезна информация об используемом алгоритме шифрования. На сегодняшний день практически для каждого криптоалгоритма существует ряд специальных методик вскрытия:

- метод бумеранга;
- сдвиговая атака;
- невозможные дифференциалы;
- атаки на связанных ключах;
- линейный криптоанализ;
- метод интерполяций;
- дифференциальный криптоанализ [2].

В статье предлагается использование нового подхода для симметричного шифрования, в котором для зашифровки открытого текста применяется особенный ключ, содержащий не только информацию о классическом ключе шифрования, но и блок данных об используемых алгоритмах. Как показано на рис. 1, новый ключ также включает суперпозицию методов.

Современные симметричные криптосистемы применяют ключи различной длины: ГОСТ 28147-89 – 256 бит; AES – 256 бит; TripleDES – 168 бит; RC5 – 128 бит; SHACAL – 512 бит; Serpent – 256 бит; BEAR– 256 бит; Kuznechik – 256 бит [2 – 4].

Применение нового подхода для формирования криптографического ключа позволит повысить криптостойкость системы, усложнив работу криптоаналитика.

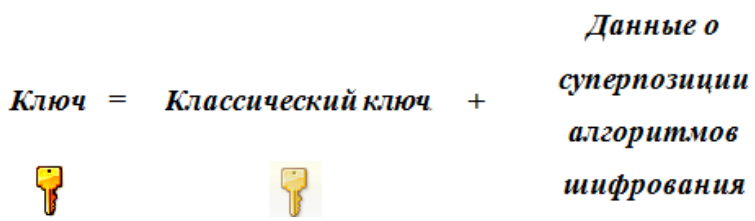

Рис. 1. Формирование особого криптографического ключа

На рис. 2 посредством диаграммы IDEF представлена схема шифрования для нового подхода. На вход поступает открытый текст. Ключ является элементом управления, так как содержит не только информацию о шифровании, но и дополнительный блок данных о последовательности алгоритмов, используемых при шифровании.

Рис. 2. Шифрование, включающее применение суперпозиции алгоритмов

Важно отметить, что алгоритмы могут быть применены не только последовательно, но и параллельно.

При *параллельном режиме* можно разбивать открытый текст на области и выполнять зашифровку каждой области с помощью разных алгоритмов и их суперпозиций.

Параллельный режим шифрования обладает следующими достоинствами:

- увеличивает криптостойкость;
- увеличивает скорость.

К недостаткам использования параллельного режима стоит отнести:

– повышение вероятности ошибки расчётов;

– дополнительные требования к аппаратному обеспечению.

Криптографические карты шифрования

Подобно электрическим схемам с параллельным и последовательным расположением элементов, можно создавать карты шифрования. Только в качестве элементов вместо составляющих цепи в криптографических картах выступают алгоритмы и их суперпозиции.

Пример карты шифрования представлен на рис. 3.

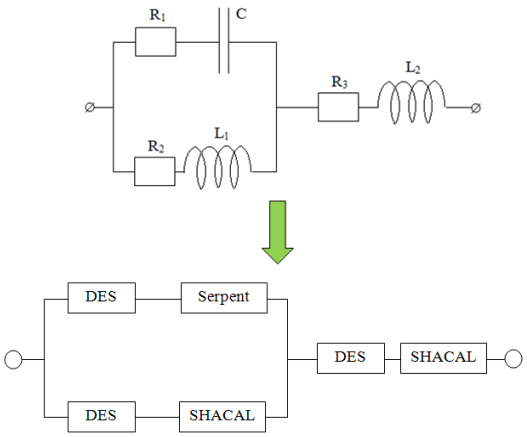

Рис. 3. Пример криптографической карты шифрования

Именно криптографические карты шифрования будут составлять дополнительный блок информации в ключе.

Для каждого алгоритма суперпозиции может быть использован один и тот же классический ключ, однако можно применять одну из процедур расширения ключа. Также можно выполнять процедуру расширения ключа в зависимости от конкретного алгоритма [2].

Модель программно-аппаратного комплекса

Модульная структура программно-аппаратного комплекса представлена на рис. 4.

Как показано на рис. 4, на входе система получает открытый текст и классический ключ.

Далее происходит *формирование криптографической карты*. Она формируется в двух вариантах:

– генерируется автоматически;

– пользователь системы сам задаёт криптографическую карту с помощью дополнительного программного обеспечения.

Как только криптографическая карта готова, выполняется процедура *преобразования классического ключа:* добавляется блок информации о суперпозиции используемых алгоритмов в виде криптографической карты.

Важно отметить, что нет необходимости дополнительного шифрования для засекречивания криптографической карты: карта содержится вместе с классическим ключом, который должен оставаться в тайне. Таким образом, секретность карты обеспечивается за счёт тайны классического ключа.

Далее осуществляется процедура *шифрования*, в ходе которой система обращается к *библиотеке алгоритмов шифрования*.

На выходе система предоставляет зашифрованный текст.

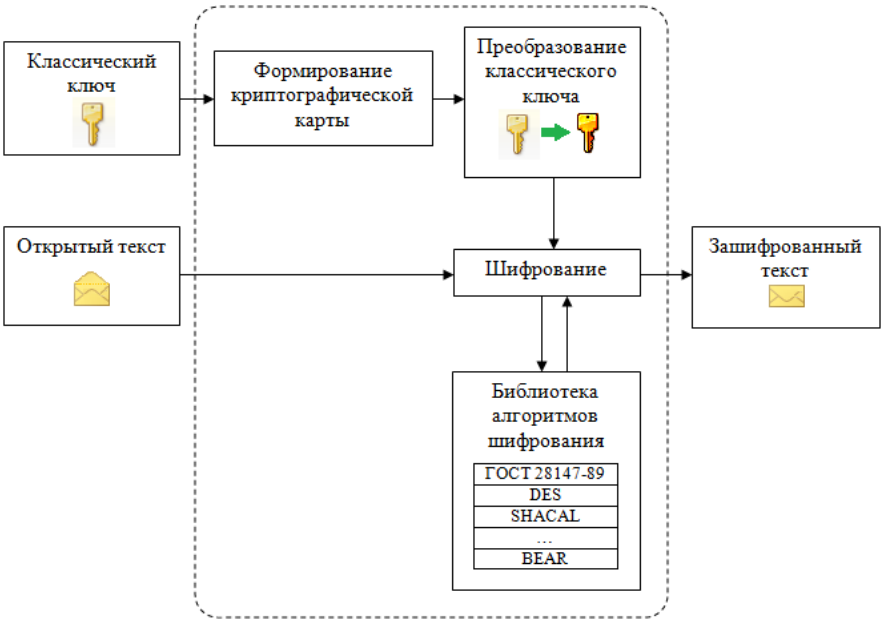

Рис. 4. Модульная структура программно-аппаратного комплекса шифрования

Формирование криптографической карты

Как отмечалось ранее, карта может быть сформирована как пользователем, так и автоматически.

Автоматическая генерация карты подразумевает существование специального механизма выбора оптимального варианта.

Для формирования карты в ручном режиме пользователю необходимо предоставить удобный графический интерфейс.

Алгоритм формирования неклассического ключа шифрования

На рис. 5 представлен алгоритм формирования ключа шифрования, состоящего

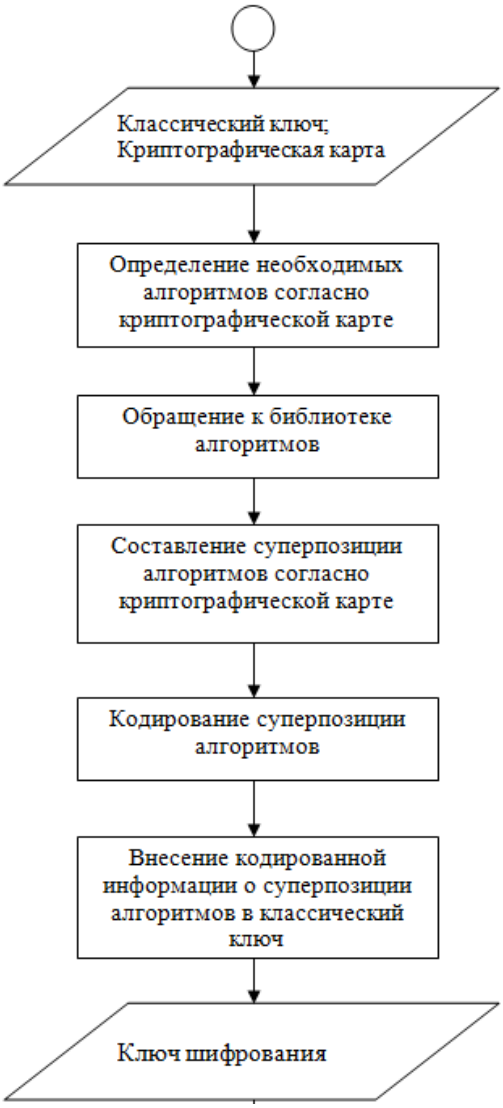

Рис. 5. Блок-схема алгоритма формирования неклассического ключа

из классического ключа и дополнительного блока информации в виде криптографической карты.

На этапе кодирования необходимо учитывать, что криптографическая карта должна быть восстановима.

Как показано на рис. 5, все действия происходят последовательно. Важно отметить, что в случае ошибки управление передаётся в конец алгоритма.

Пример интерфейса программы формирования криптографической карты в ручном режиме

На рис. 6 представлен интерфейс для составления криптографической карты пользователем.

Как показано на рис. 6, слева расположена область проектирования карты, справа – панель используемых алгоритмов. Список алгоритмов можно редактировать с помощью соответствующего управляющего элемента – кнопки «Добавить».

Важно отметить, что криптографическая карта не должна храниться в отдельном файле. Данное положение повышает уровень информационной безопасности, а также снимает дополнительные требования к объему постоянной памяти. Криптографическая карта должна быть включена в классический ключ и храниться только в нём.

Кодирование криптографической карты в классическом ключе шифрования

Внесение блока информации о суперпозиции используемых алгоритмов симметричного шифрования не должно влиять на основные параметры классического ключа. Необходимо стремиться к тому, чтобы злоумышленник не смог догадаться, что ключ содержит дополнительные данные, а также к тому, чтобы ключ не терял параметры безопасности.

Безопасность алгоритма сосредоточена в ключе, и если использовать криптографически слабый процесс для генерации ключей, то система в целом будет слабо защищена [1]. В связи с этим необходимо предусмотреть оптимальный принцип кодирования суперпозиции криптографических алгоритмов, в классический ключ.При кодировании важно выполнять следующие требования:

– компактное размещение блока информации в ключе;

– незаметное размещение блока информации в ключе;

– минимальное влияние на параметры безопасности ключа.

Таким образом, предлагаемая в статье криптографическая система обладает гибкостью: в неё могут входить как проверенные алгоритмы, так и совершенно новые (в зависимости от политики формирования библиотеки) [5].

Требования к аппаратному обеспечению

Важно отметить, что при распараллеливании любого алгоритма необходимо заранее позаботиться о требованиях к вычислительным ресурсам. Чем больше операций будет одновременно выполняться, тем большее количество процессоров понадобится [6].

Безусловно, идеальным вариантом будет применение вычислительных систем, использующих протокол MPI [7]. Однако подобного рода системы требуют больших материальных ресурсов.

Разбиение на алгоритмы может происходить в автономном режиме, при котором система сама решает, какой набор алгоритмов будет применен и в какой последовательности.

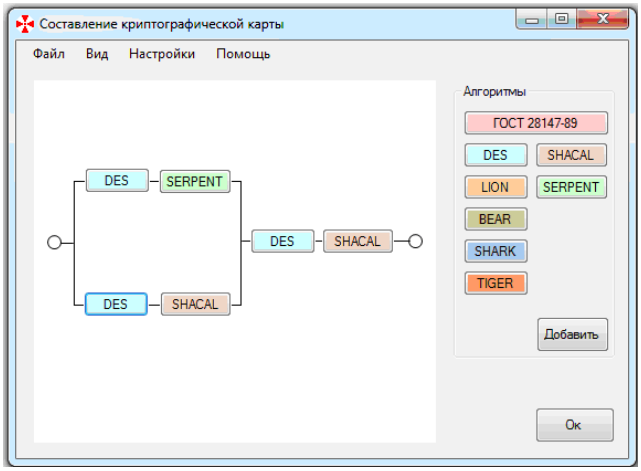

Рис. 6. Интерфейс программы формирования криптографической карты в ручном режиме

Система использует множество алгоритмов, поэтому необходимо учитывать объем оперативной памяти.

Главным достоинством предложенного подхода является то, что он представляет собой гибкую систему управления алгоритмами шифрования, в которой для шифрования одного открытого текста подразумевается использование нескольких алгоритмов.

СПИСОК ЛИТЕРАТУРЫ

1. Шнайер, Б. Прикладная криптография. Протоколы, алгоритмы и исходящие тексты на языке Си /Б. Шнайер. – 2-е изд. – М.: Триумф, 2012. – 816 с.
2. Панасенко, С. Алгоритмы шифрования. Специальный справочник / С.Панасенко. – СПб.: БХВ-Петербург, 2009. – 578 с.
3. Хорев, П.Б. Методы и средства защиты информации в компьютерных системах: учеб. пособие для студентов высш. учеб. заведений /П.Б.Хорев.– 4-е изд., стер. – М.: Академия, 2008. – 256 с.
4. ГОСТ Р 34.12 – 2015. Информационные технологии. Криптографическая защита информации. Блочные шифры. – М.: Стандартинформ, 2015.
5. Карлова, Т.В. Разработка концепции обеспечения многоуровневого доступа к конфиденциальной информации / Т.В. Карлова, Н.М. Кузнецова // Вестник МГТУ «Станкин». – 2011. – № 2 (14). – С. 87-90.
6. Кузнецова, Н.М. Применение усовершенствованного криптоаналитического метода «грубой силы» в автоматизированной системе разграничения доступа к конфиденциальной информации /Н.М. Кузнецова, Т.В. Карлова // Вестник МГТУ «Станкин». – 2012.– №4(23). – С. 139-143.
7. Немнюгин, С.А. Программирование для многопроцессорных вычислительных систем / С.А. Немнюгин. – СПб.: БХВ-Петербург, 2013. – 400 с.

1. Scheier, B., Applied cryptography. *Transactions, Algorithms, and Outgoing Texts in C Language* / B. Schneier. – 2nd ed. – M.: Triumph, 2012. – pp. 816.
2. Panasenko, S., Encryption algorithms. *Special Reference-Book* / S. Panasenko. – S-P.: BKhV-Petersburg, 2009. – pp. 578.
3. Khorev, P.B., Methods and means of information security in computer systems: *Text-Book for Students of Colleges* /P.B. Khorev. – 4th ed. Stereotyped. – M.: Academy, 2008. – pp. 256.
4. SARS 34.12 – 2015. *Information Techniques. Information Cryptographic Security. Block Ciphers*. – M.: Standardinform, 2015.

5. Karlova, T.B., Development of concept for support of multi-level access to confidential information / T.B. Karlova, N.M. Kuznetsova // *Bulletin of MSTU "Stankin".* – 2011. – No 2(14). – pp. 87-90.
6. Kuznetsova, N.M., Application of advanced crypto-analytical method of "brute force" in automated system of differentiation of access to confidential information / N.M. Kuznetsova, T.B. Karlova // *Bulletin of MSTU "Stankin".* – 2012. – No 4(23). – pp. 139-143.
7. Nemnyugin, S.A., Programming for multi-processor computer systems / S.A. Nemnyugin. – S-P.: BKhV-Petersburg, 2013. – pp. 400.

Материал поступил в редколлегию 29.06.15.
Рецензент: д.т.н., профессор А.С.Верещака

Сведения об авторах:

Карлова Татьяна Владимировна, д.с.н., к.т.н., профессор, ведущий научный сотрудник Института конструкторско-технологической информатики РАН, тел.: 8-(499)-978-99-62, 8-(903)-776-90-78, e-mail: karlova-t@yandex.ru.
Кузнецова Наталия Михайловна, к. т. н., преподаватель кафедры «Автоматизированные системы обработки информации и управления» Московского государственного технологического университета «Станкин», тел.: 8 (499)- 972-94-37, 8-(903)-581-80-15, e-mail: knm87@mail.ru.
Бекмешов Александр Юрьевич, к. т. н., доцент, старший научный сотрудник Института конструкторско-технологической информатики РАН, тел.: 8-(499)-978-99-62, 8-(926)-582-34-35, e-mail: b-a-y-555@yandex.ru

Karlova Tatiana Vladimirovna,D.S., Can.Eng., Prof., Leading Researcher Institute of Design-Technological Informatics of RAS, Phone: 8-(499)-978-99-62, 8-(903)-776-90-78, e-mail: karlova-t@yandex.ru.
Kuznetsova Natalia Mikhailovna, Can.Eng., Lecturer of the Dep. « Automated Systems for Information processing and Control » Moscow State Technological University «Stankin », Phone : 8 (499)- 972-94-37, 8-(903)-581-80-15, e-mail: knm87@mail.ru.
Bekmeshov Alexander Yurievich, Can.Eng., Senior Researcher Institute of Design-Technological Informatics of RAS, Phone: 8-(499)-978-99-62, 8-(926)-582-34-35, e-mail: b-a-y-555@yandex.ru.